

What is claimed is:

1. A multiprocessor system comprising:

a plurality of processors which send and receive predetermined information to and from each other; and

a shared memory which is shared and accessed by said plurality of processors one
5 after another, and

wherein each of said plurality of processors requests at least one of other processors included in said plurality of processors to access said shared memory that is to be done by the at least one of other processors, in a case where each of said plurality of processors has accessed said shared memory.

2. The multiprocessor system according to claim 1, wherein each of said plurality of processors

requests, in a case where each of said plurality of processors has normally updated predetermined data in said shared memory, the at least one of other processors to read
5 same data as the predetermined data from said shared memory.

3. The multiprocessor system according to claim 1, wherein each of said plurality of processors

requests, in a case where each of said plurality of processors has not normally updated predetermined data in said shared memory, the at least one of other processors to
5 update the predetermined data in said shared memory.

4. A multiprocessor system comprising:

a plurality of processors which send and receive predetermined information to and from each other;

a shared memory which is shared and accessed by each of said plurality of
5 processors; and

an access manager which manages access to said shared memory by each of said plurality of processors, and

10077947.022002

wherein said access manager selects, in a case where said plurality of processors are in contention to access said shared memory, one of said plurality of processors being in
10 contention and permits the selected processor to access said shared memory, and

each of said plurality of processors requests at least one of other processors included in said plurality of processors to access said shared memory that is to be done by the at least one of other processors, in a case where each of the processors is selected by said access manager and the selected processor accesses said shared memory.

5. The multiprocessor system according to claim 4, wherein each of said plurality of processors

requests, in a case where each of said plurality of processors has normally updated predetermined data in said shared memory, the at least one of other processors to read
5 same data as the predetermined data from said shared memory.

6. The multiprocessor system according to claim 4, wherein each of said plurality of processors

requests, in a case where each of said plurality of processors has not normally updated predetermined data in said shared memory, the at least one of other processors to
5 update the predetermined data in said shared memory.

7. The multiprocessor system according to claim 4, wherein each of said plurality of processors

requests, in a case where a predetermined period of time has elapsed without being selected by said access manager, the at least one of other processors and said access
5 manager to perform a predetermined reset operation for resetting themselves.

8. A multiprocessor system comprising:

a plurality of processors which send and receive a predetermined signal to and from each other;

a shared memory which is shared and accessed by each of said plurality of
5 processors; and

a contention determiner which detects whether said plurality of processors are in contention to access said shared memory, and permits one of said plurality of processors to access said shared memory, and

wherein each of said plurality of processors outputs a access-request signal to at least one of other processors included in said plurality of processors, so as to request the at least one of other processors to access said shared memory, in a case where each of said plurality of processors is permitted to access said shared memory by said contention determiner and the permitted processor accesses said shared memory.

9. The multiprocessor system according to claim 8, wherein each of said processors

outputs, in a case where each of said processors accesses said shared memory and has normally updated predetermined data therein, a re-read request signal to the at least one of other processors, so as to request the at least one of other processors to read same data as the predetermined data from said shared memory.

10. The multiprocessor system according to claim 8, wherein each of said plurality of processors

outputs, in a case where each of said plurality of processors accesses said shared memory and has not normally updated predetermined data therein, a update-request signal to the at least one of other processors, so as to request the at least one of other processors to update the predetermined data in said shared memory.

11. The multiprocessor system according to claim 8, wherein each of said plurality of processors outputs a reset-request signal to the at least one of other processors and said contention determiner, so as to request the at least one of other processors and said contention determiner to reset themselves, in a case where a predetermined period of time has elapsed without being selected by said access manager.

12. A shared-memory controlling method to be executed in a multiprocessor system including a plurality of processors which send and receive predetermined

10077947-022002

information to and from each other, a shared memory which is shared and accessed by each of said plurality of processors, and an access manger which manages access to said 5 shared memory by each of said plurality of processors, and said method comprising:

selecting one processor included in said plurality of processors, and permitting the selected one processor to access said shared memory, in a case where said plurality of processors are in contention for said shared memory;

performing first access to said shared memory using the selected processor;

10 requesting at least one of other processors included in said plurality of processors to perform second access to said shared memory, in a case where said performing the first access to said shared memory has been done; and

performing the second access to said shared memory using the at least one of other processors.

13. The shared-memory controlling method according to claim 12, wherein said requesting includes requesting, in a case where predetermined data has normally been updated in said performing the first access, the at least one of other processors to read the predetermined data from said shared memory.

14. The shared-memory controlling method according to claim 12, wherein said requesting includes requesting, in a case where predetermined data has not normally been updated in said performing the first access, the at least one of other processors to update the predetermined data in said shared memory.

15. The shared-memory controlling method according to claim 12, wherein said requesting includes requesting, in a case where a predetermined period of time has elapsed without being selected in said selecting, the at least one of other processors and said access manager to perform a predetermined reset operation for resetting 5 themselves.

16. A shared-memory controlling method comprising:

selecting one of a plurality of processors, and permitting the selected processor to

10077947.022002

access a shared memory shared by the plurality of processors, in a case where the plurality of processors are in contention for the shared memory;

- 5 performing first access to said shared memory using the selected processor;
requesting at least one of other processors included in said plurality of processors to perform second access to said shared memory, in a case where the first access has been done; and

performing the second access to said shared memory using the at least one of other
10 processors.

17. A computer readable recording medium for controlling a computer to execute a shared-memory controlling method comprising:

- selecting one processor included in a plurality of processors, and permitting the selected one processor to access a shared memory, in a case where said plurality of
5 processors are in contention for said shared memory;

performing first access to said shared memory using the selected processor;
requesting at least one of other processors included in said plurality of processors to perform second access to said shared memory, in a case where said performing the first access to said shared memory has been done; and
10 performing the second access to said shared memory using the at least one of other processors.

18. A data signal embedded in a carrier wave and representing an instruction sequence for controlling a computer to execute a shared-memory controlling method comprising:

- selecting one of a plurality of processors, and permitting the selected processor to
5 access a shared memory shared by the plurality of processors, in a case where the plurality of processors are in contention for the shared memory;

performing first access to said shared memory using the selected processor;
requesting at least one of other processors included in said plurality of processors to

perform second access to said shared memory, in a case where the first access has been
10 done; and

performing the second access to said shared memory using the at least one of other
processors.

10077947.022002